

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
6 June 2002 (06.06.2002)

PCT

(10) International Publication Number
WO 02/44977 A1

(51) International Patent Classification⁷: G06F 17/60, 9/46

(21) International Application Number: PCT/US00/32689

(22) International Filing Date:
30 November 2000 (30.11.2000)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicant (for all designated States except US):
NETSCAPE COMMUNICATIONS CORPORATION [US/US]; 501 E. Middlefield Road, Mountain View, CA 94043 (US).

(72) Inventors; and

(75) Inventors/Applicants (for US only): **MANSOUR, Steve** [US/US]; 2215 Wellington Drive, Milpitas, CA 95035 (US). **GRAVES, Mikol** [US/US]; 2468 Funston Avenue, San Francisco, CA 94116 (US).

(74) Agents: **GLENN, Michael et al.**; Glenn Patent Group, 3475 Edison Way, Suite L, Menlo Park, CA 94025 (US).

(81) Designated States (national): AE, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CR, CU, CZ, DE, DK, DM, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA.

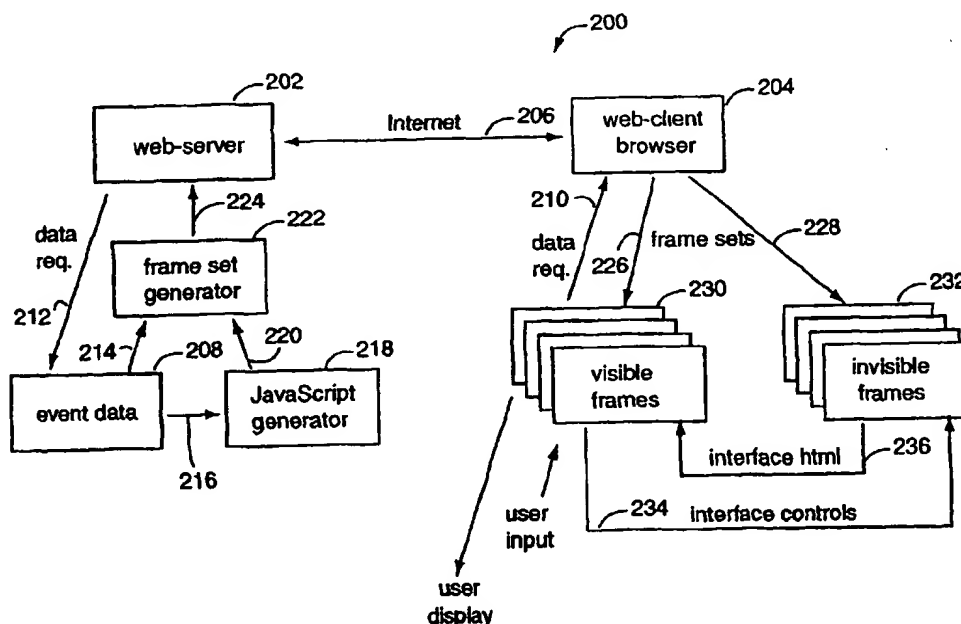
(84) Designated States (regional): ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published:

— with international search report

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: JAVASCRIPT CALENDAR APPLICATION FOR INTERNET WEB BROWSER



(57) Abstract: The present invention is a calendaring system implemented as a JavaScript application for program execution on individual Internet browsers after being downloaded by a webserver. The JavaScript application generates HTML on-the-fly and a graphical user interface is displayed on a user's screen. The result is an interactive scheduling system that can be shared between users on the Internet.

WO 02/44977 A1

JAVASCRIPT CALENDAR APPLICATION FOR INTERNET WEB BROWSER

5

BACKGROUND OF THE INVENTION

TECHNICAL FIELD

10

The present invention relates to web browsers and the Internet, and more specifically to calendar client applications that can run on all computer platforms and that improve calendar-server scalability.

15

DESCRIPTION OF THE PRIOR ART

The success of the Internet means that each web server will be hit on by more and more web browsers. It therefore follows that there is less CPU time available at the web server to handle each visitor. So allowing more than one iteration between a users and server for a single result is a luxury than can no longer be afforded. It also now makes sense to transfer responsibility for any needed processing from the web server to the web browser.

Netscape Calendar is a program that allows users to manage their time more efficiently by maintaining a calendar of a user's activities. Users can place items on the calendar as needed in order to stay organized. Netscape Calendar is intended for collaborative use, so each user can access the calendars of other users and plan meetings or other events without phone calls or e-mail messages. Netscape Calendar is a part of Netscape Communicator Professional Edition and needs to be activated by an administrator before it can be used. But such conventional system requires far too much support and attention from the web server.

Netscape Calendar includes Agenda, a sharable calendar, tasks, daily notes, and daily events. Netscape Agenda also provides access to the user's tasks, daily notes, day events and reminders. Any event scheduled in a user's

- agenda's day, week or month view is an agenda entry. Users can create and view a user's agenda entries and, in some cases, those of others, depending on a user's access rights and the access level the creator has assigned the entry. Users may also edit any entries created in a user's name. Users can
- 5 create tasks in a user's agenda. Tasks are things users have to do, but that cannot be scheduled into a user's agenda like a meeting or an appointment. Such tasks appear in a task view of a user's daily agenda pages and in a user's task display. Users can create daily notes in a user's agenda. Notes are entered into a user's agenda, not already entered under tasks, day events, or
- 10 agenda entries. The daily notes appear in the notes view of a user's daily and weekly agenda pages. A day event lasts for an entire day, without taking up time in a user's day view. Day events will appear in the notes view at the bottom of a user's agenda pages.
- Netscape Calendar can remind users of the entries users have in it. Such
- 15 reminders can be set up in a number of different ways, to suit the demands of a user's entries and a user's schedule. Users can print out a user's agenda pages. The different types of pages can be tailored to include only the information users want on a user's printouts. The fonts and margins can also be adjusted to suit a user's needs.
- 20 Each Netscape Calendar server manages a database of individual calendars, the number of which is limited by the capabilities of the server hardware. As with POP/IMAP e-mail mailboxes, individual calendars always sit on the same servers, so each calendar has a "home" server. Calendars also may also exist on a user's local system, and can select any number of calendars to be
- 25 synchronized onto the local system. When a suitable network connection is available, the local calendars can be synchronized with a server-based calendar database. Users typically synchronize only their own calendar, but Netscape Calendar servers support making local, synchronized copies of any calendar in a system. A user who is traveling could therefore bring along a whole
- 30 department's calendars. When multiple calendars exist on the same calendar server, free/busy searches can be run on that server. When user's calendars are spread across many servers, a user's home server must connect separately to each server to gather the free/busy information and to present a unified view.
- 35 United States Patent 5,960,406, issued September 28, 1999, to Rasansky, et al., describes a scheduling system for use between users on the web. Each

end user is granted a unique password protected personal calendar. This calendar is generated from information stored in a database at a central server, and delivered to each end user as standard HTML sent through the Internet. This custom personal calendar is then viewed by the end user in a standard Web Browser. This obviates the need for special software programs to be purchased by end users, and also allows end users of any CPU type to read their calendars. When an end user uses the system to send an invitation or announcement to others on the system, the sending end user has the option of sending e-mail in addition to posting that information in the calendars of others.

When an end user sends an invitation or announcement to a person who is not an Appointnet user, then the Appointnet system automatically creates a unique calendar for the recipient, and sends an e-mail to that person. Individuals who use the present system can post reminders to themselves, send announcements to people they know, and make appointments with people they know. When these messages are sent, the communication is nearly instantaneous because the system makes one record and allows both (or many) parties to view it. Such Patent is incorporated herein by reference.

SUMMARY OF THE INVENTION

The present invention is a calendaring system implemented as a JavaScript application for program execution on individual Internet browsers after being downloaded by a webserver. The JavaScript application generates HTML on-the-fly from within invisible frames and renders such HTML on a user's screen in visible frames. The result is an interactive scheduling, appointment, and calendaring system that can be shared between many users on the Internet.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 is a functional block diagram of an Internet calendaring system embodiment of the present invention; and

Fig. 2 is a dataflow diagram of a calendaring system embodiment of the present invention in which a web-server sends frame sets that open in users' browsers as visible and invisible frames.

DETAILED DESCRIPTION OF THE INVENTION

5 Fig. 1 is a functional block diagram of a calendar system embodiment of the present invention referred to herein by the general reference numeral 100. The Internet 102 is used to interconnect network servers and clients. A calendar webserver 104 provides a shared calendar control and synchronization function for many clients distributed about the Internet. Such clients and users can share
10 and exchange calendar information to help coordinate community events, private meetings, classroom attendance, legal deadline observance, etc. An output 106 transmits hypertext transfer protocol (HTTP) datapackets that include calendar core routines written in JavaScript, for example, reference user's interfaces (ref-UI) written in hypertext markup language (HTML), and
15 calendar event data. Such are issued in response to requests and event data received in HTTP datapackets on an input 108. A number of typical web-clients and their browsers are represented by web-clients 110, 112, and 114. The users as such may be independent or loosely associated in a variety of groups and special interests. A remote webserver 116 can include a sponsor
20 who pays a fee to the operator of the calendar webserver 104 to include commercial advertisements in the JS-core, ref-UI, and calendar event data on output 106.

The JS-core, ref-UI, and calendar event data on output 106 are received on an input 118 to web-client 110 in response to requests issued to the calendar
25 webserver 104 over an output 120. The web-client 110 may generate original calendar event data that will be stored in the calendar server and can be distributed to the appropriate users by the calendar webserver 104. Similarly, requested JS-core, ref-UI, and calendar event data on output 106 are received on an input 122 to web-client 112 in response to requests issued to the
30 calendar webserver 104 over an output 124. This web-client 112 may also generate unique and original calendar event data that needs to be distributed to the appropriate users by the calendar webserver 104. The web-client 114 is no different. An input 126 receives JS-core, ref-UI, and calendar event data on output 106. An output 128 handles requests to calendar webserver 104 and
35 any special event data generated. An output 130 from the remote webserver provides advertisements and content in typical HTML and JavaScript http-

datapackets. Any requests, e.g., hyperlinks clicked on by users at the web-clients 110, 112, and 114, are received on an input 132.

The web-clients' browsers 110, 112, and 114 must support frames, e.g., multiple windows that can be generated and controlled by the webserver 104.

5 For example, Netscape Navigator 3.0, or later version, will be preferred. The JS-core, ref-UI, and calendar event data on output 106 will initially cause a frame set to be created. Some of the frames in the frame set will be visible to the web-client users, and some will not be visible. Those that are not visible are used to interface the event data on the webserver with the visible frames on
10 the web-client browser. The ref-UI is coded in Javascript within HTML and will build a graphical user interface in a model-calendar format, e.g., days, weeks, months. The JS-core is coded in JavaScript and provides interactive users control locally, e.g., from within one of the non-visible frames. An initial download of event data from the webserver to the web-client will be
15 preferably adequate to service most if not all read-only calendar interactions by a user. Any missing or needed event data will be requested as needed. New, original event data generated by a user that is important for other users to have is uploaded to the webserver 104. Changes to existing event data are uploaded as well.

20 A web-client user interface is included in the calendar server. A web-client user interface is generated "on-the-fly" within a user's browser. First, a frame set is created. Several frames are visible to the user, and several frames are "hidden" and not visible on the screen. One such frame includes calendar event data that was requested from the server. Other frames include JavaScript
25 routines that know how to read the event data and produce HTML to render the events or other interface elements. The JavaScript running in the hidden frames emit HTML to the visible frames to render the interface seen by the users. As the user presses links and controls on the interface, calls are made into the JavaScript routines to change the interface. For example, if the user presses
30 the month button, the JavaScript routines will emit a monthly calendar view of data and send it to the visible frames.

One advantage to this approach is that a lot of processing is done in a user's browser, and a round trip to the server is not required for every button click. For example, suppose that several months worth of event data is downloaded in
35 one call to the server. Suppose that the user is viewing one day's worth of data. Now the user clicks the "next day" button. It is likely that the event data

for the next day is already in the invisible data frame. Assuming it is, the JavaScript routines detect this, and emit the HTML for the next day and display it. There was no need to make a round trip request to the server. The number of requests that the server must process is reduced because many requests
5 can be processed on the web browser by the JavaScript. Furthermore, for users with phone modem connections to the ISP, the new page can be generated much faster than a web page can be transmitted from the server.

Such interface generation allows links to images, ads, or other content to be accessed from a remote server.

10 The web-client code is a combination of JavaScript and HTML. It is divided into two major parts, a JS-core and a reference user-interface (UI). The JS-core includes routines to: (a) fetch, edit, create and delete calendar events, (b) login/logout, (c) import/export calendar information, (d) preference management routines, e.g., agenda list management, initial view, first day of week, time-zone,
15 (e) calendar management routines (WCAP commands), (f) localize strings, (g) change locales, (h) set colors and fonts, (i) set themes (specific combinations of colors, fonts, and logo images), (j) format dates and time zones, and (k) date navigation, date utilities, and interface utilities.

The reference user-interface implements a calendar user interface based on the
20 JS-core routines. Such includes linked events, agendas (layers of calendars), and, public and private calendars. Users can own multiple calendars, and calendars can be owned by multiple users. Links can be embedded in web pages or e-mail messages to point to individual events or calendar views. E-mail alarms, e-mail paging, and e-mail invitations to events are also supported
25 by the calendar web server 104. User preferences typically include preferred first-day-of-week, preferred time zone, multiple time zone support, import/export/synchronization, print preview, deletion tombstones, color scheme, font scheme, sound scheme, and context-sensitive help.

Fig. 2 illustrates a calendaring system embodiment of the present invention,
30 and is referred to herein by the general reference numeral 200. The system 200 is based on a web-server 202 that services at least one web-client 204 over an Internet connection 206. A calendar event database 208 stores coordinated, corrected, and up-to-date calendar information in condensed form. A data request 210 initiated by a browser user at a network client site includes a
35 description of what particular calendar information a user wants. This is

forwarded over the Internet 206 and becomes a data request 212. The appropriate data is fetched and its presentation may require certain user interfaces to deal with it.

Therefore, the calendar event database 208 responds to queries with an event data 214 and an event-interface description 216. A JavaScript generator 218
5 builds corresponding JavaScript routines 220 that will be executed as-needed by the web-client 204. A frame set generator 222 builds a mixed event data, HTML, and JavaScript code 224 for transmission to the web-client 204.

At the web-client 204, such mixed event data, HTML, and JavaScript code is
10 separated into a visible-frames data 226 and an invisible-frames data 228. A frames-capable browser responds with a set of visible frames 230 that appear before the user and a set of invisible frames 232. For example, the visible frames 230 can include day, week, month, and year interactive graphical user interfaces for appointment, event, and schedule data of concern to the user.

15 The purpose of the invisible frames 232 is to host the downloaded JavaScript routines and calendar data 220. A user-interface control 234 will trigger various ones of the JavaScript routines to execute and generate new user-interface HTML 236 that will render within or build more visible frames 230.

In alternative embodiments of the present invention, a "standalone" or "native"
20 client is needed that has off-line capabilities, sync capabilities, and is feature rich. Traditionally, these clients also had to be developed on multiple platforms (Windows, Mac, and Unix). Such calendar client preferably has entirely downloadable chrome, i.e., the entire user interface look-and-feel can be downloaded to a client that understands a description language such as
25 XML/CSS. It should look, feel, and act like a native client, and actually be an application that makes use of browser/XML/CSS technology.

Some embodiments of the present invention preferably can incorporate attachments to events. A back-end that supports an iCalendar GEO property (geographic event location), is exposed in the interface. Meeting locations are
30 tied to mapping services to allow users to obtain personalized maps and directions to event locations. Layout management tools are preferably included for customizing the interface. Automated operations include adding an extra frame on the top, bottom, or side of each window, and adding links to web address on each page.

A fully functional calendaring system preferably incorporates portions of Netscape Messaging Server. Such enables users to exchange information within a company and across the Internet. Messaging Server is controlled by electronic mail or HTML forms and lets administrators manage users information and system-configuration parameters with the easy-to-use, point-and-click interface of Netscape Navigator and Communicator from any desktop on the network. It offers feature richness without compromising messaging interoperability or standards compliance.

Messaging Server version-3.5 provides numerous feature enhancements over the previous releases, including: Support for Internet Message Access Protocol Version-4 (RFC 1730) to provide messaging support for remote users, including support for IMAP4rev1 (RFC 2060) for optimal performance of message throughput. E.g., integration with the latest release of the frames-based administration of Netscape SuiteSpot 3.1 for centralized administration of all Netscape servers; procedures for doing bulk additions, deletions, and modifications that allow quick migration of existing users. Integrated NIS and NIS+ lookup capability is useful to facilitate address resolution outside of Messaging Server's domain.

The SSL 3.0 support in Netscape Messaging Server administration is used for secure remote administration and client communications, and LDAP version-3 support (RFC 2251) for centralized users management, message routing, and international character sets. Authenticated SMTP (to prevent unauthorized Message transmissions) and IMAP over SSL (to fully encrypt communications between the server and the client) are important. Support for delivery status notifications, to determine status of sent messages inside or outside the corporation, and improved network manageability via SNMP and NT EventVwr and Perfmon are included. Support is needed for messaging Internet Foundation Classes, and for creating mail-enabled applications between the client and server. A server application programming interface (API) helps to develop customized transport-enable applications.

Messaging Server supports the Lightweight Directory Access Protocol (RFC 1777) for managing its user's information and for routing messages. Messaging Server interoperates with a wide variety of third-party directory tools and Netscape Directory Server. Messaging Server automatically creates, deletes, or changes the account when it receives an update. Messaging Server uses an account database provided by any LDAP-compliant directory server. IMAP4

is based on work by the University of Washington and is embodied in the RFC 1730 specification. It allows users to be disconnected from the main messaging system and still be able to process their mail. The specification allows for administrative controls for these disconnected users and for the
5 resynchronization of the user's message store once the user reconnects to the messaging system.

IMAP4 as an open standard does allow for the integration of security mechanisms for the client authentication to the messaging server. An encrypted messaging transport protocol is not part of the IMAP4 specification and has
10 been developed to the S/MIME standard in Netscape Communicator.

Although the invention is preferably described herein with reference to the preferred embodiment, one skilled in the art will readily appreciate that other applications may be substituted for those set forth herein without departing from the spirit and scope of the present invention. Accordingly, the invention
15 should only be limited by the Claims included below.

CLAIMS

1. A calendaring system method, the method comprising the steps of:
5 creating a frame set in which there are a plurality of visible frames and a plurality of invisible frames;
 including a calendar event data that was requested from the server in said frame set;
 transmitting at least one frame in said plurality of invisible frames with a
10 JavaScript routine able to read said calendar event data and able to generate HTML-code for rendering events and interface elements;
 generating HTML-code within one of said plurality of invisible frames that renders within one of said plurality of visible frames a user interface; and
 calling said JavaScript routine to change said user interface in response to
15 a user clicking-on a variety of links and controls rendered in said user interface;
 wherein, a user interface is generated within a web client "on-the-fly" within a user's browser.
2. The method of claim 1, wherein:
20 the step of creating a frame set is instigated from a webserver and executed by said user's browser.
3. The method of claim 1, further comprising the steps of:
 storing and maintaining a database of events and tasks at a webserver;
25 and
 including a selected and relevant portion of the database of events in the step of creating a frame set.
4. The method of claim 3, wherein:
30 the step of including is in-response to an event-data request sent from said user's browser to said webserver.
5. The method of claim 1, wherein:
 the step of generating HTML-code within one of said plurality of invisible
35 frames is a result of executing a particular JavaScript routine supplied by said webserver and hosted by said user's browser in one of said invisible frames.

6. The method of claim 1, further comprising the steps of:
inserting advertising for display in one of said visible frames.
7. The method of claim 6, further comprising the steps of:
5 sending advertising information for said display from a remote server that is independent of said webserver and a web-client.
8. A calendaring system, comprising:
a webserver having an Internet connection;
10 a web-client having a browser and network connection to said Internet;
an event database included in the webserver and providing for the storage and maintenance of appointment, calendar, task, and event information relevant to at least one user;
a JavaScript generator included in the webserver and providing for
15 JavaScript routines targeted to execute in the web-client by said browser;
a frame-set generator included in the webserver and providing for a transmission of event data, HTML for frames, and JavaScript to the web-client and said browser;
a plurality of visible frames generated by the web-client and said browser
20 and displayed to said user; and
a plurality of invisible frames generated by the web-client and said browser and not displayed to said user;
wherein said JavaScript routines are eventually hosted in at least one of the plurality of invisible frames and when executing produce HTML-code that is
25 rendered in at least one of the plurality of visible frames.
9. The calendaring system of claim 8, wherein:
the plurality of visible frames constitute displays of user appointments and event calendars, and a user can interact with them to change calendar display
30 formats and time periods.
10. The calendaring system of claim 8, wherein:
the web-client allows a user to initiate an event-data request that is answered by the webserver with said transmission of event data, HTML for
35 frames, and JavaScript to the web-client and said browser.

1/2

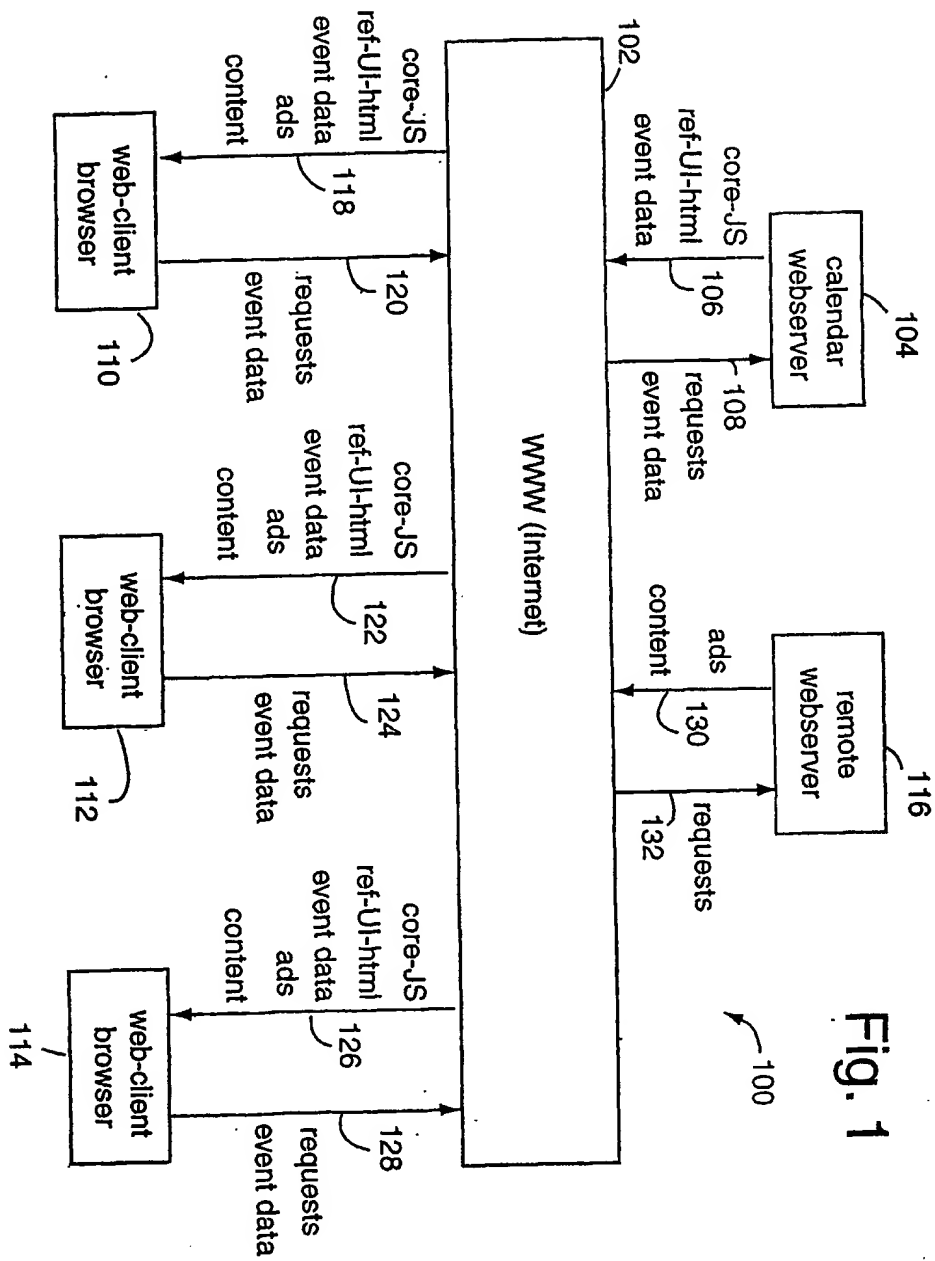
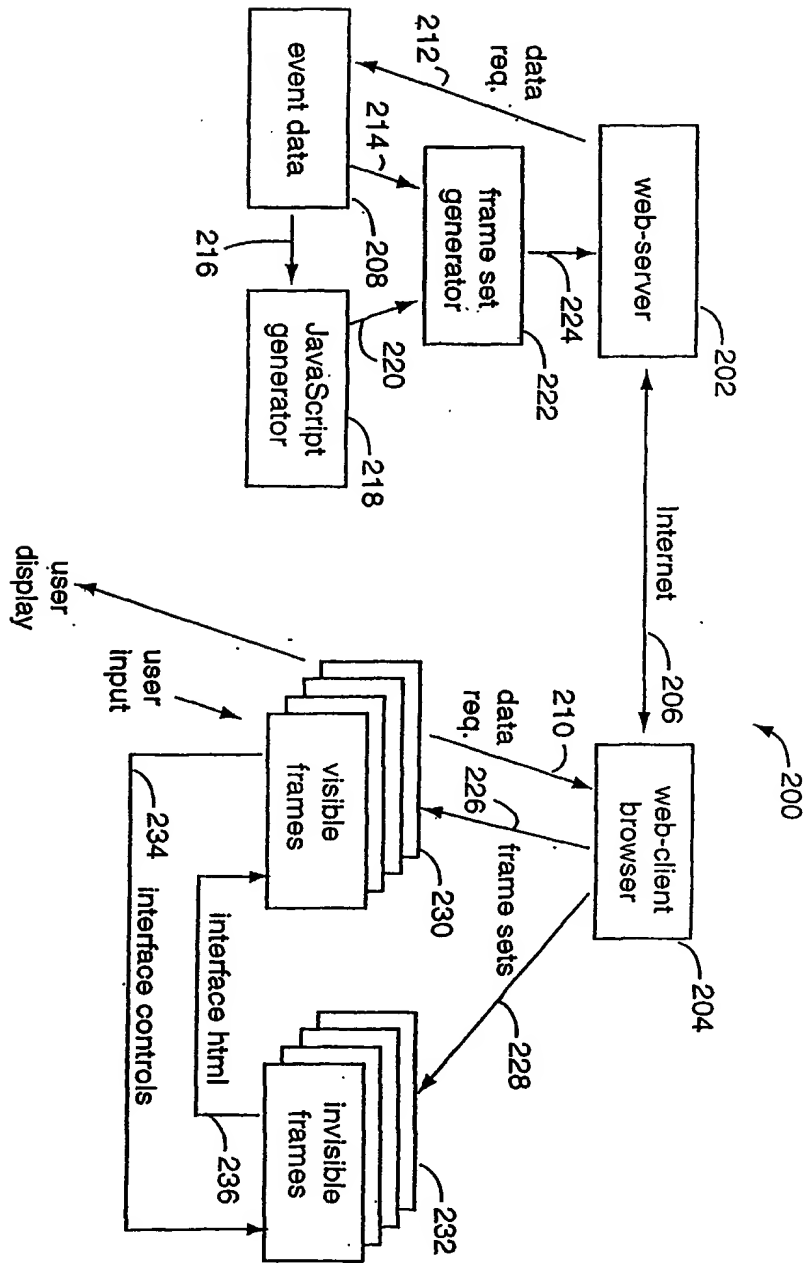


Fig. 1

Fig. 2



INTERNATIONAL SEARCH REPORT

International Application No
PCT/JP 00/32689

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 G06F17/60 G06F9/46

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the International search (name of data base and, where practical, search terms used)

EPO-Internal, WPI Data, PAJ, IBM-TDB

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 5 960 406 A (DENTON III LEET E ET AL) 28 September 1999 (1999-09-28) cited in the application abstract; figure 1A column 1, line 54 -column 2, line 58 — -/-	1-10

☒ Further documents are listed in the continuation of box C.

☒ Patent family members are listed in annex.

* Special categories of cited documents :

- *A* document defining the general state of the art which is not considered to be of particular relevance
- *E* earlier document but published on or after the International filing date
- *L* document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
- *O* document referring to an oral disclosure, use, exhibition or other means
- *P* document published prior to the International filing date but later than the priority date claimed

T later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

X document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

Y document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

Z document member of the same patent family

Date of the actual completion of the International search

28 November 2001

Date of mailing of the International search report

05/12/2001

Name and mailing address of the ISA
European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,
Fax: (+31-70) 340-3016

Authorized officer

Pinheiro, T

INTERNATIONAL SEARCH REPORT

International Application No
PCT/US 00/32689

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT		
Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	<p>PALMER ET AL: "Web-based tools in support of life cycle engineering of telecommunications products"</p> <p>ELECTRONICS MANUFACTURING TECHNOLOGY SYMPOSIUM, 1997., TWENTY-FIRST IEEE/CPMT INTERNATIONAL AUSTIN, TX, USA 13-15 OCT. 1997, NEW YORK, NY, USA, IEEE, US, 13 October 1997 (1997-10-13), pages 134-139, XP010249717</p> <p>ISBN: 0-7803-3929-0</p> <p>abstract; figure 2</p> <p>page 135, column 2, line 4 - line 32</p> <p>page 137, column 1, line 6 -column 2, line 3</p> <p>page 138, column 2, line 12 - line 36</p>	1-10
A	<p>TRECORDI V ET AL: "AN EXPERIMENTAL STUDY ON THE PERFORMANCE OF WWW SERVERS"</p> <p>COMMUNICATIONS: THE KEY TO GLOBAL PROSPERITY. GLOBECOM 1996. GLOBAL INTERNET 96 CONFERENCE RECORD. LONDON, NOV. 18 - 22, 1996, GLOBAL TELECOMMUNICATIONS CONFERENCE (GLOBECOM), NEW YORK, IEEE, US, vol. SUPP, 18 November 1996 (1996-11-18), pages 22-27, XP000741667</p> <p>ISBN: 0-7803-3337-3</p> <p>abstract</p> <p>page 22, column 1, line 33 - line 37</p> <p>page 24, column 2, paragraph 3.1.6</p>	1-10
A	<p>WO 00 43898 A (ECONVENE INC ;SMITH WILLIAM DEAN (US)) 27 July 2000 (2000-07-27)</p> <p>abstract</p> <p>page 14, line 3 - line 6</p>	1-10

INTERNATIONAL SEARCH REPORT

International Application No
PCT/US 00/32689

Patent document cited in search report		Publication date	Patent family member(s)		Publication date
US 5960406	A	28-09-1999	AU	2239299 A	09-08-1999
			EP	1049983 A1	08-11-2000
			WO	9938079 A1	29-07-1999
WO 0043898	A	27-07-2000	AU	2854800 A	07-08-2000
			WO	0043898 A1	27-07-2000